Macromedia Director MX 2004

47 Navigation Scripts

In this lesson you will:

- Attach Scripts to Cast Members
- Reuse a Single Behaviour to Manage Button States

Important note for lesson:

Before you begin this lesson you should have downloaded the file 14_02. zip, which contains all the resource files needed for this lesson.

When downloading the accompanying zip files and extracting resource files onto your computer, please note that the storage location may differ to that shown in lesson graphics. Names of folders and file structures may be different as you have the choice of where to store them on your computer.

When you begin building scripts, you may be tempted to write complete scripts from beginning to end, including all of the variables, calls to custom handlers and so on. However, it would be very unlikely that you could create such scripts without errors. Instead, you will start with simple scripts, testing each one to be sure it works before moving on to more complex scripts.

In this lesson we will embed Lingo code in a Cast member and experience how it executes. You will then replace your Cast member code with a behaviour from the Library Palette.

Points to remember:

One of the decisions you will have to make before you begin programming is how the scripts will be attached to the various Sprites on the Stage. You can create a re-usable behaviour script that you will have to attach to each button in turn or you can embed the script in a Cast member. If you choose to embed a script, each instance of that Cast member will contain the script.

For this lesson

- Open the file **Profiler.dir**

We will begin by reviewing the reusable behaviours that have already been assigned to the navigation Sprites.

- Open the **Behaviour Inspector**

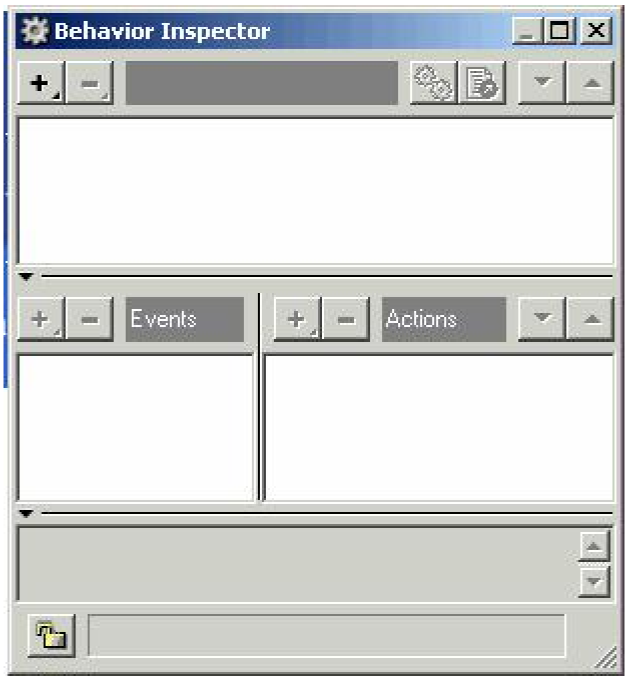
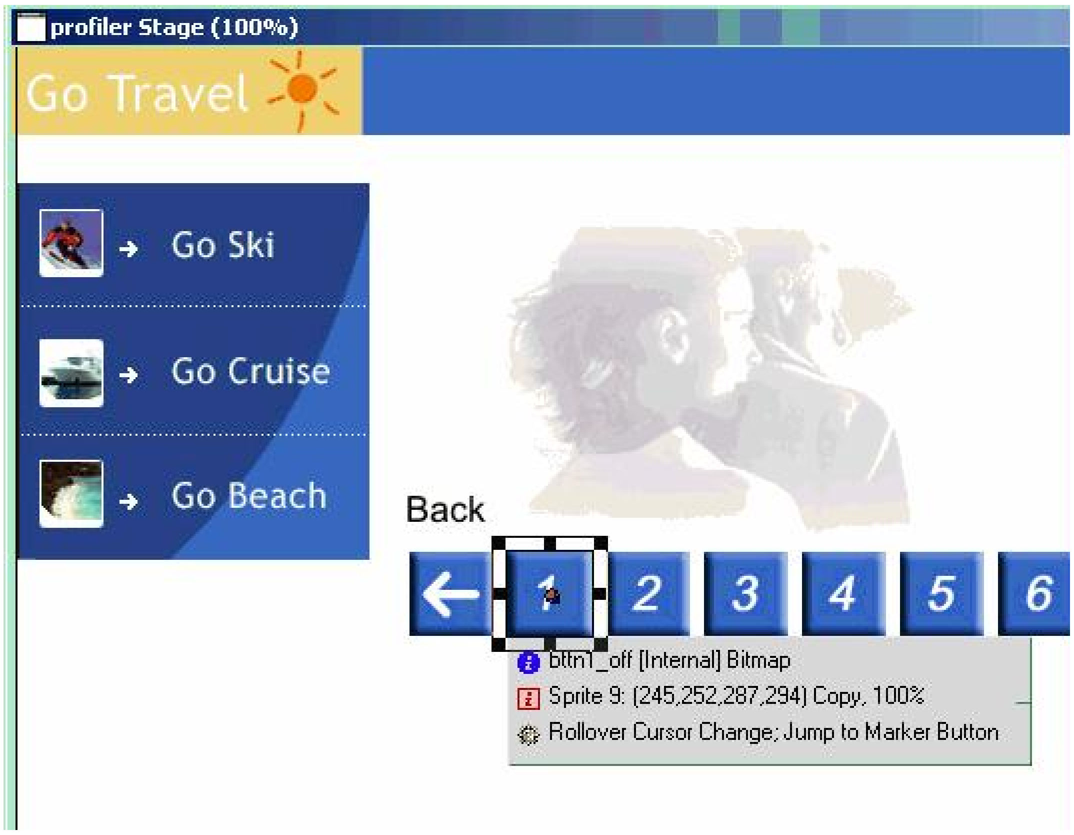


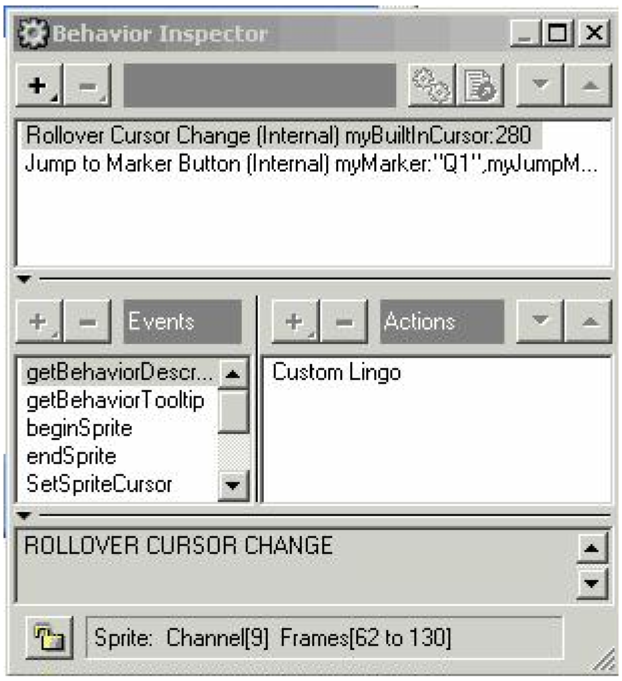
Diagram 1: Behaviour Inspector

- Click on any of the navigation Sprites displayed in the Score, in Frame **70**



bttn1_off [Internal] Bitmap
Sprite 9: (245,252,287,294) Copy, 100%
Rollover Cursor Change; Jump to Marker Button

Diagram 2: Selected navigation Sprite



Rollover Cursor Change (Internal) myBuiltInCursor:280
Jump to Marker Button (Internal) myMarker:"Q1",myJumpM...

getBehaviorDescr...
getBehaviorTooltip
beginSprite
endSprite
SetSpriteCursor

Custom Lingo

ROLLOVER CURSOR CHANGE

Sprite: Channel[9] Frames[62 to 130]

Diagram 3: Behaviour Inspector window

Notice that all the Sprites include a **RollOver Cursor Change** and **Jump to Marker Button** behaviours.

- Click on Sprite **faded image**, at Frame **65 channel 1**.

We will play the movie to evaluate the current navigation.

- Close the **Behaviour Inspector**
- Close the **Score** window
- Click on the **Play** button

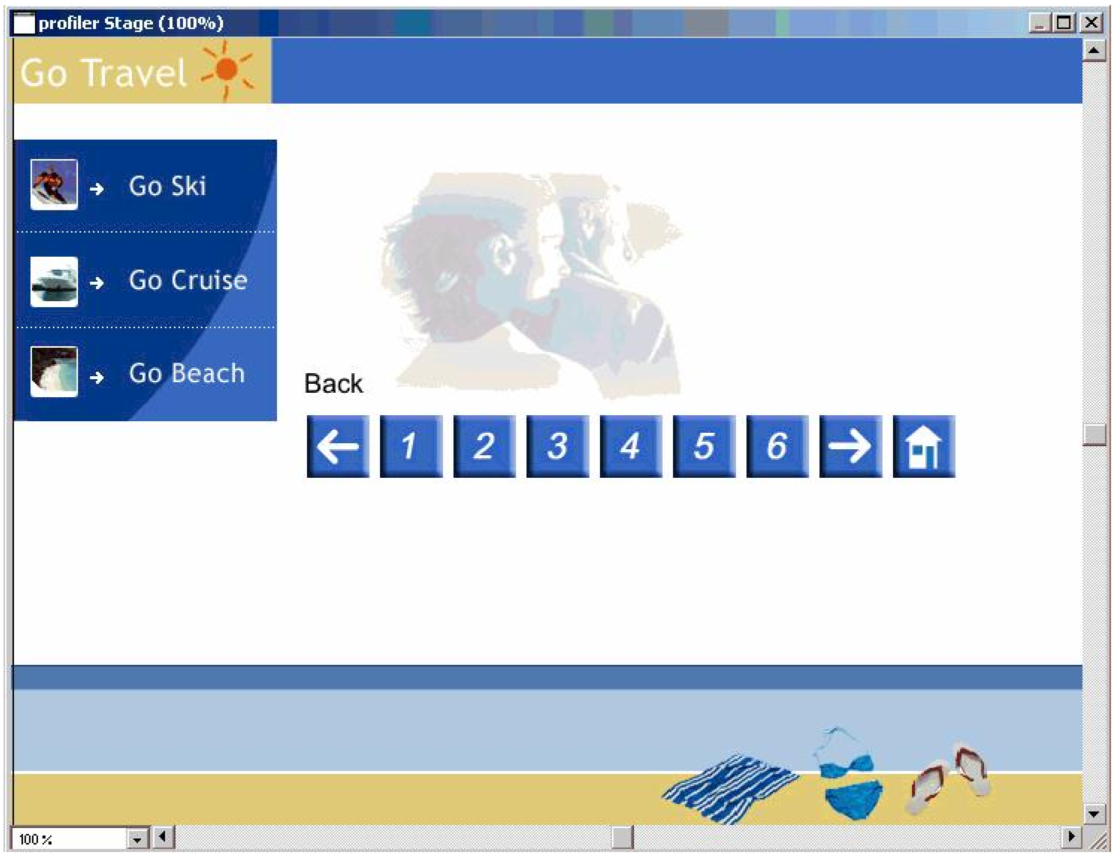


Diagram 4: Screenshot of the stage

Click on any of the Navigation buttons. Notice that you can jump to any question, as well as to the Home, Help and Result pages. Notice too that the buttons do not give the user any feedback. The buttons do not move when they are clicked on, nor do they change when the mouse moves over them.

- Click on the **Stop** button

The Playback head has been positioned over the Home page. We will add functionally to the Home button by attaching a script to the appropriate Cast member.

- Open the Cast window

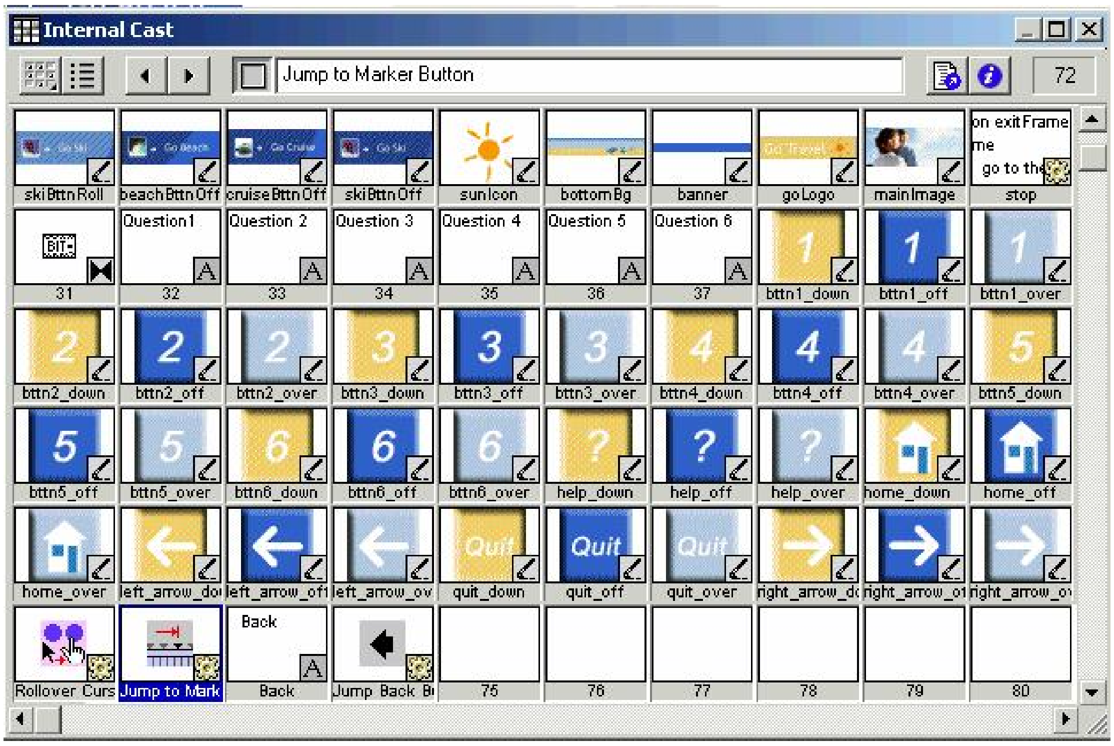


Diagram 5: Cast Window

- Select **59:home_down** and select **61:home_over**

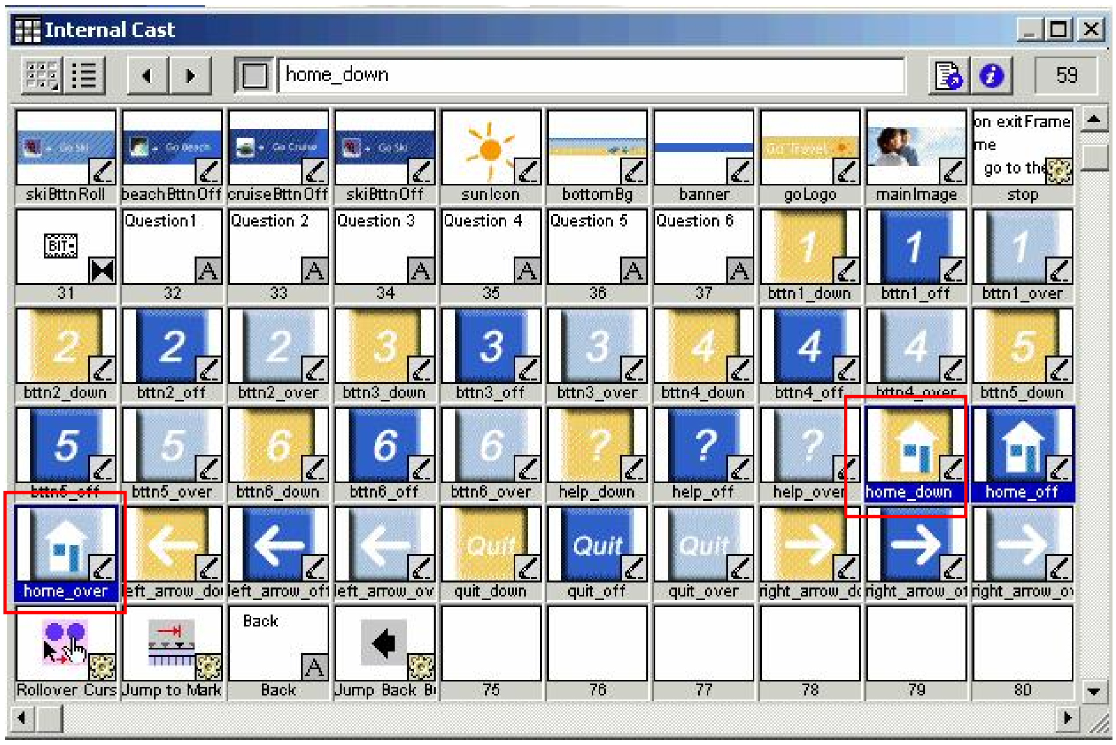


Diagram 6: Selected home_down and home_over

- Select **60:home_off**

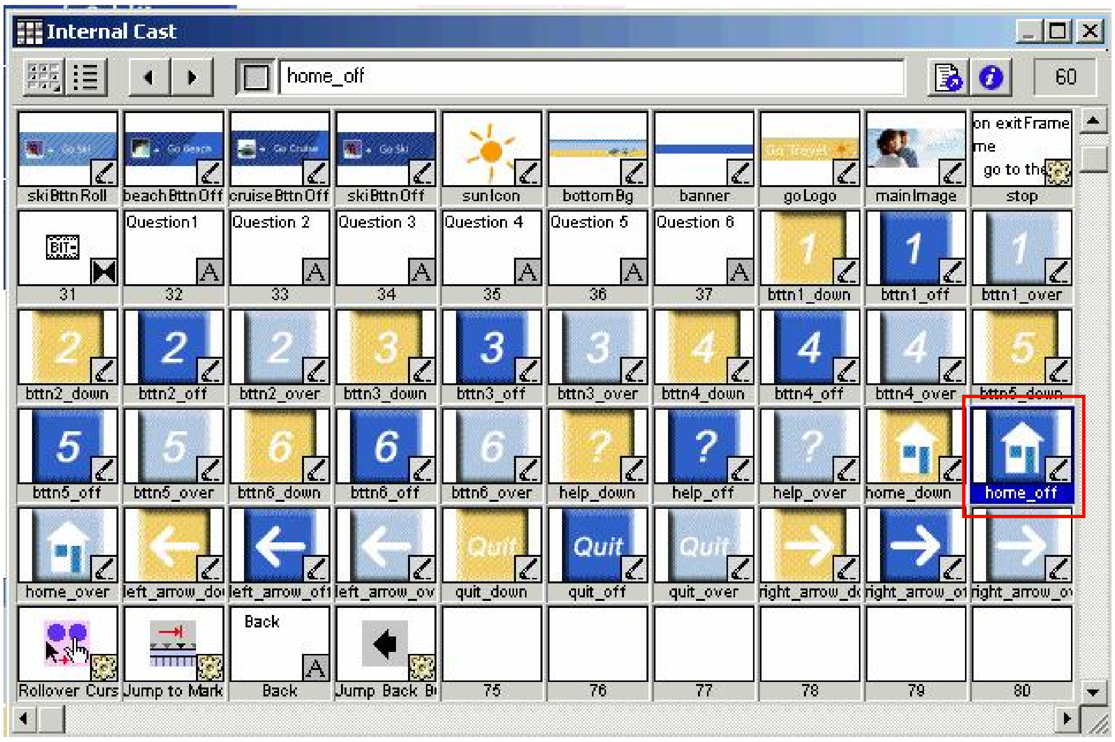


Diagram 7: home_off

- Click on the **Cast Member Script** button

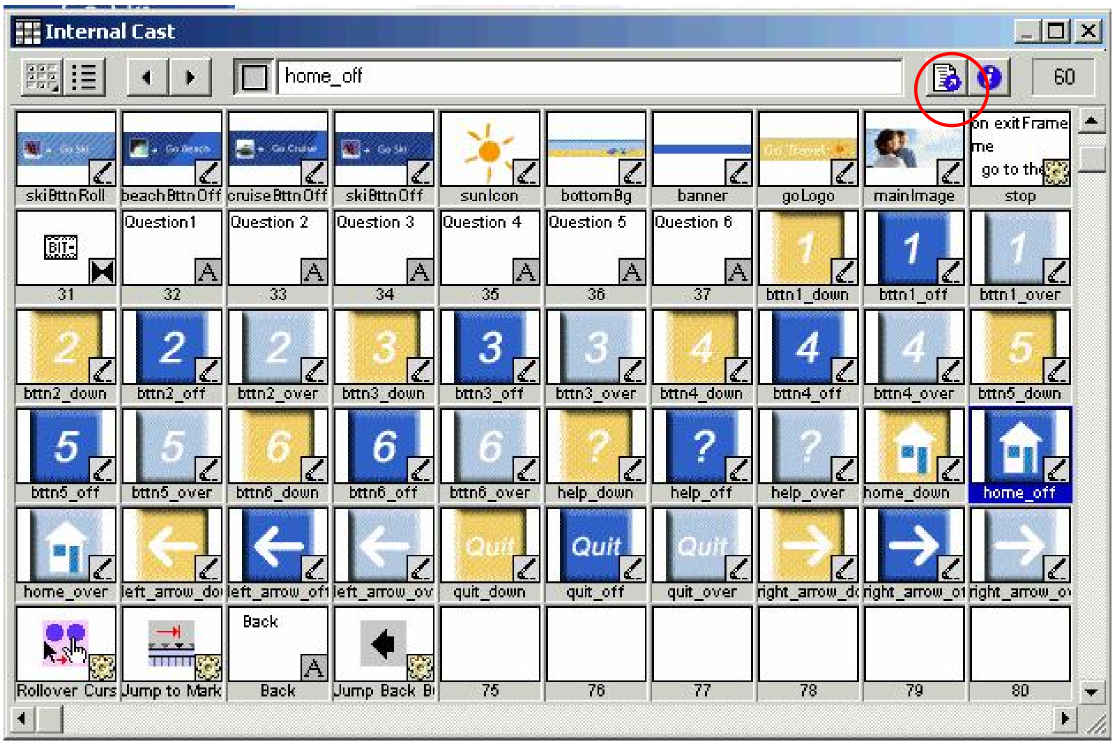
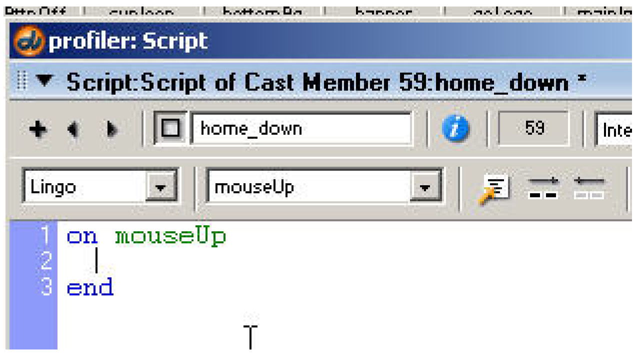


Diagram 8: Cast member script button



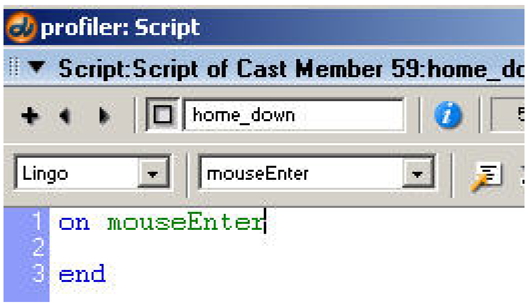
```
on mouseUp
  
end
```

Diagram 9: Script window

As you observed, the Cast contains a down and over state, as well as the normal state, for the Home button. We will write the Lingo script that will display the over state image when the mouse enters the area of the Home button on the Stage.

- Double-click on the event name **mouseUp**

- Type **mouseEnter** and press **Enter**



profiler: Script

Script:Script of Cast Member 59:home_dc

home_down

Lingo | mouseEnter

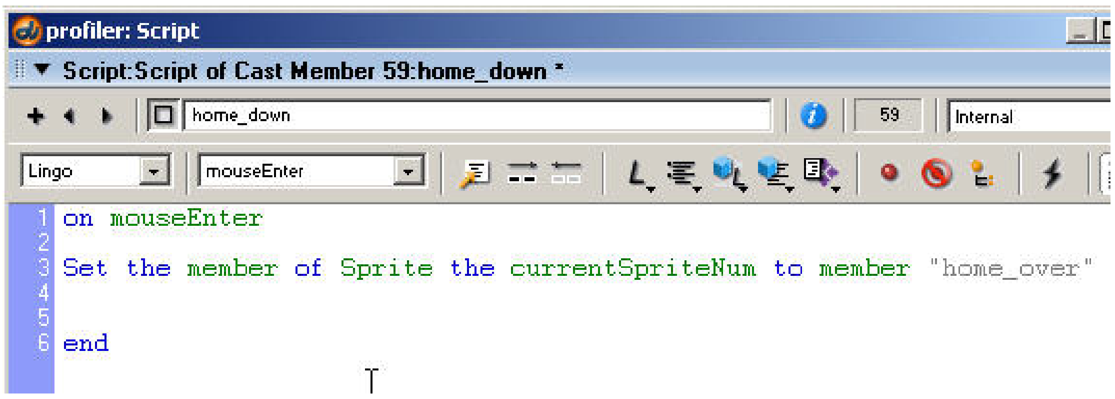
```
on mouseEnter

end
```

Diagram 10: mouseEnter

You can enter the Lingo code for the mouseEnter event handler.

- Type the highlighted code below

Set the member of Sprite the currentSpriteNum to member "home_over"



profiler: Script

Script:Script of Cast Member 59:home_down *

home_down | 59 | Internal

Lingo | mouseEnter

```
on mouseEnter

Set the member of Sprite the currentSpriteNum to member "home_over"


end
```

Diagram 11: Script window with added code

This Lingo script will replace the **Home_off** Cast member with the **Home_over** Cast member when the mouse moves over or enters the area of the Home cast member.

- Close the **Script** window

Unlike when you create behaviours, no additional Cast member appears in the Cast window. However, a script icon appears on the 60:home thumbnail, indicating that a script is attached.

- Click on the **Cast Window** button
- Click on the **Play** button
- Move the mouse over the **Home** button

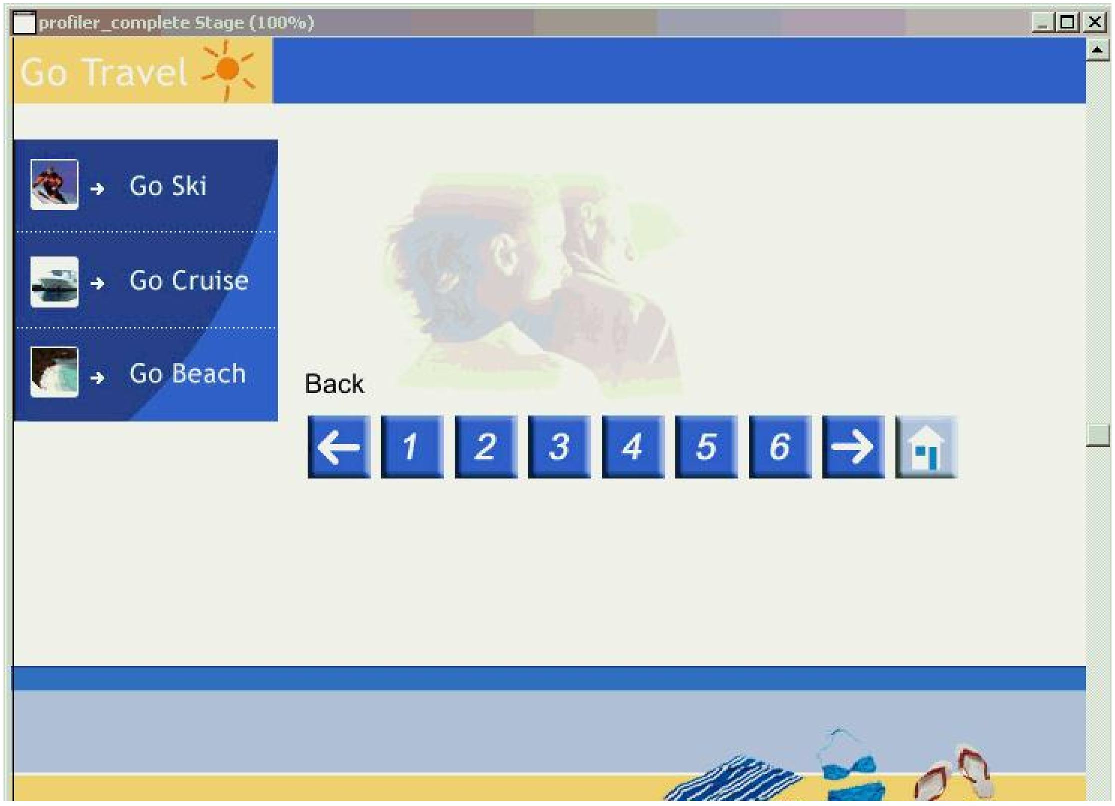
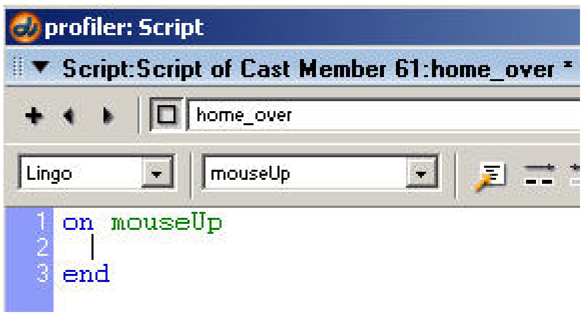


Diagram 12: Screenshot of mouse over the Home button

The **home_off** cast member in the Sprite has been replaced with the **home_over** cast member.

To restore the Home image to the Sprite, you could write a cast member script in the home_over Cast.

- Stop the movie
- Open the Cast window and select **61:home_over**
- Click on the **Cast Member Script** button



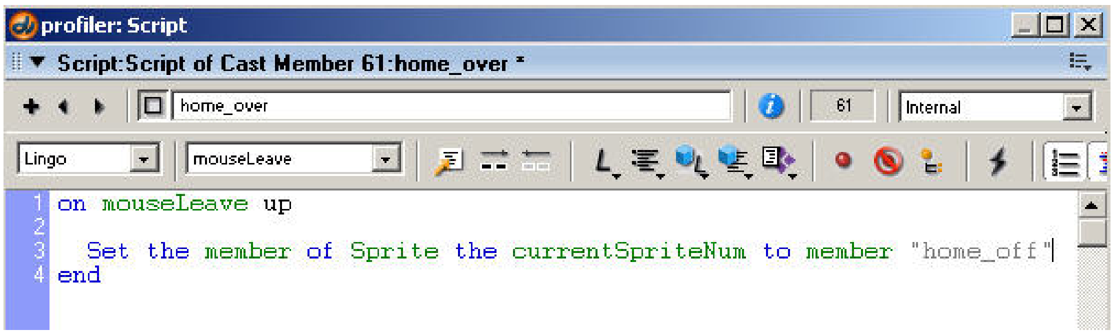
```
on mouseUp

end
```

Diagram 13: Script window

We will add the Lingo script to restore the Home cast member to the Sprite when the mouse leaves the area of the home_over image.

- Double-click on the event name **mouseUp**
- Type **mouseLeave** and press **Enter**
- Type the highlighted code below.

Set the member of Sprite the currentSpriteNum to member "home_off"



```
on mouseLeave up

  Set the member of Sprite the currentSpriteNum to member "home_off"
end
```

Diagram 14: Window with script

Once again, we will play the movie to see the effect of the Cast member Lingo script.

- Close the Script Window
- Click on the **Cast Window** button to close the Cast window
- Click on the **Play** button

Test the effect of the Lingo code you have added to the two Cast members. It works, but it is also a bit overwhelming. Using this method of placing Lingo code within a cast member, you will need to write very specific scripts for several events, for each one of the buttons in the movie's navigation. We will use behaviours to do this.

- Stop the movie

A well written behaviour will enable you to assign all the image states you need. Behaviours can be assigned to the frame channel but they can also be assigned to Sprite channels.

You do not want your behaviour Lingo code bumping into the Cast member Lingo code.

- Close the **Profiler.dir file** that you are working on
- Open the **Profiler_2.dir file** located in the same resource folder as the Profiler.dir file

Behaviours can include properties and the properties can be defined wherever the behaviour is used. Using properties, behaviours become reusable Lingo scripts, making them very flexible. The behaviour you want needs to swap images of a Sprite based on different mouse events.

- Open the **Score** window
- Click on the **home_off Sprite** in channel 16**,** in frame 66

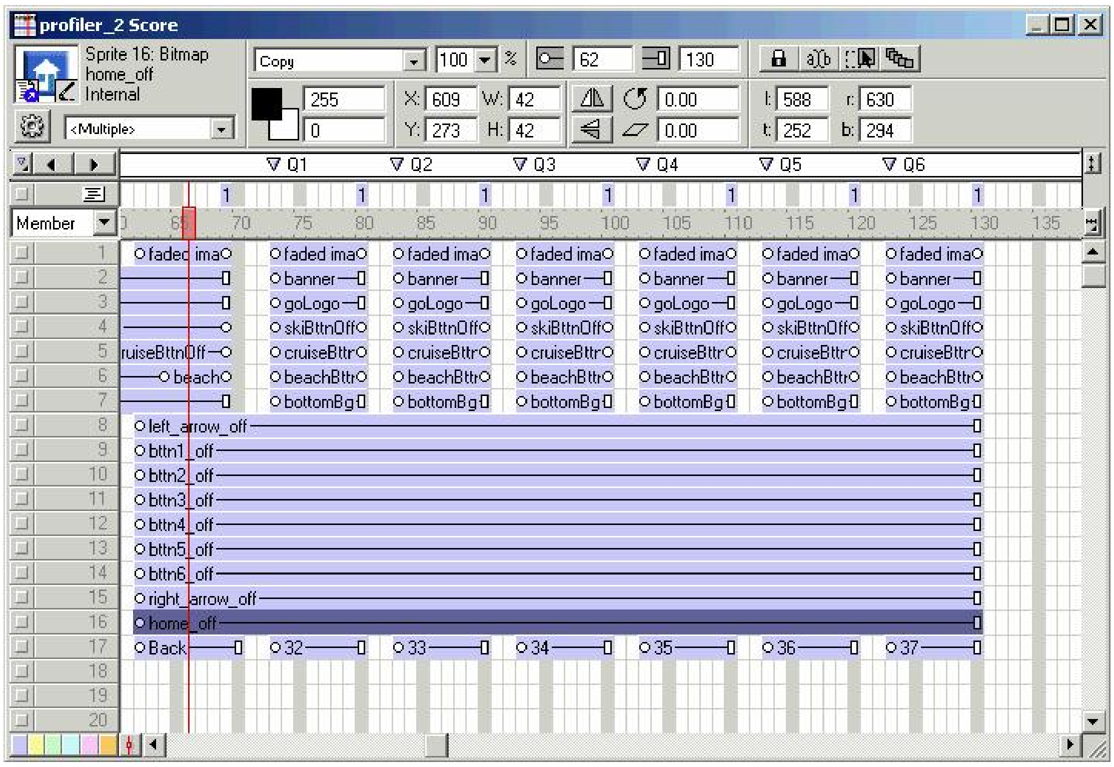


Diagram 15: Selected Sprite

- Click on the **Library Palette**

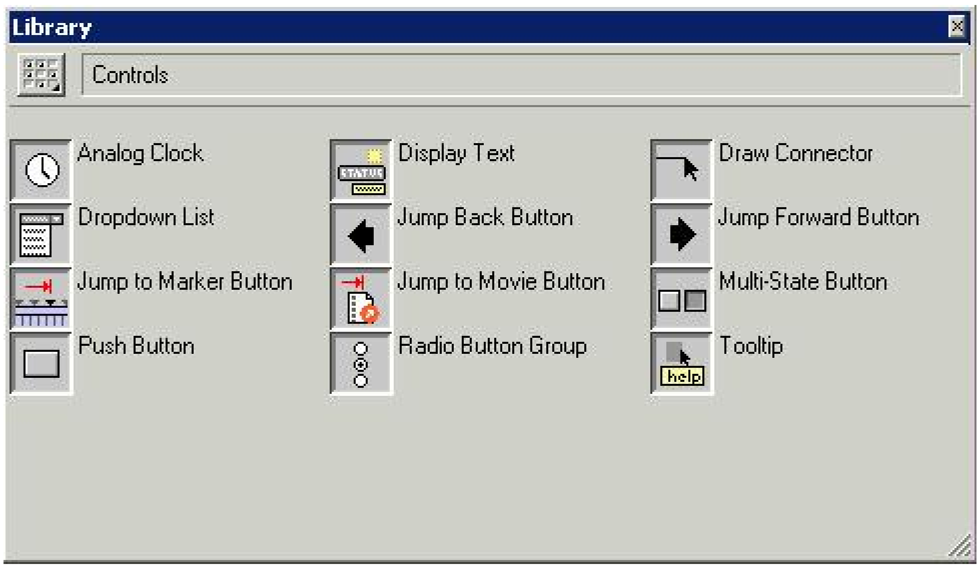


Diagram 16: Library palette

You want to use the Push Button behaviour.

- Drag the **Push Button** behaviour from the Library Palette onto the **home_off Sprite**

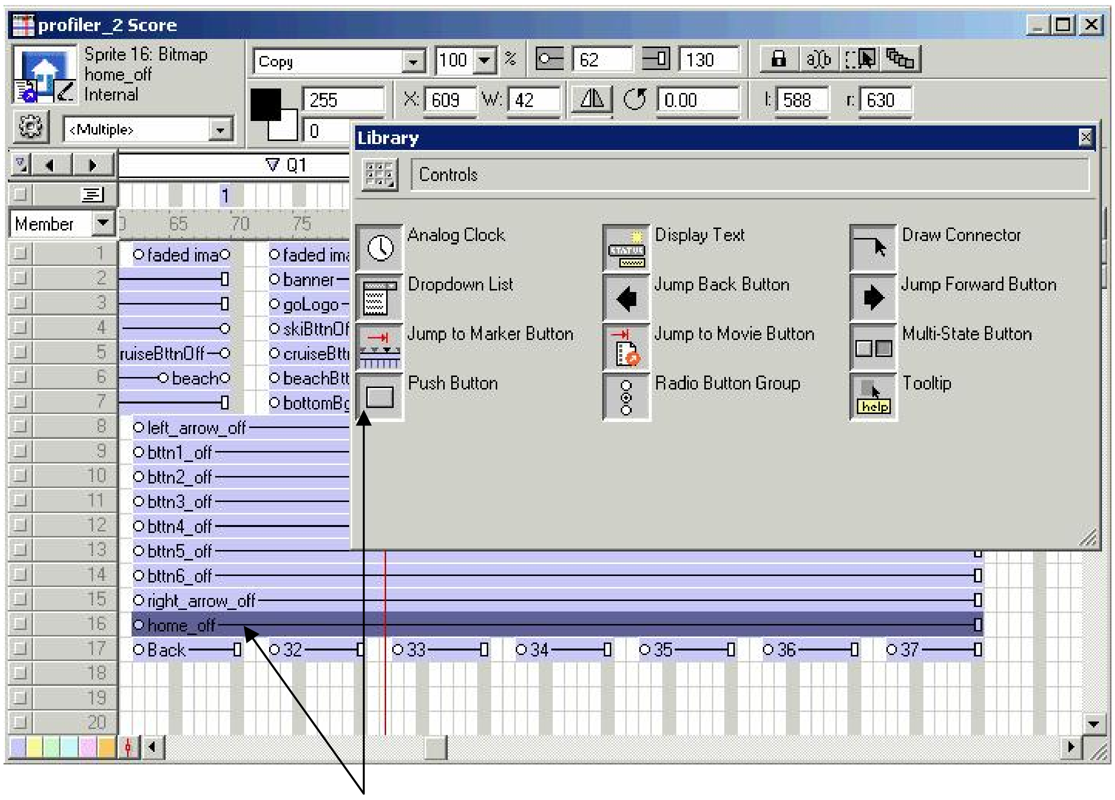
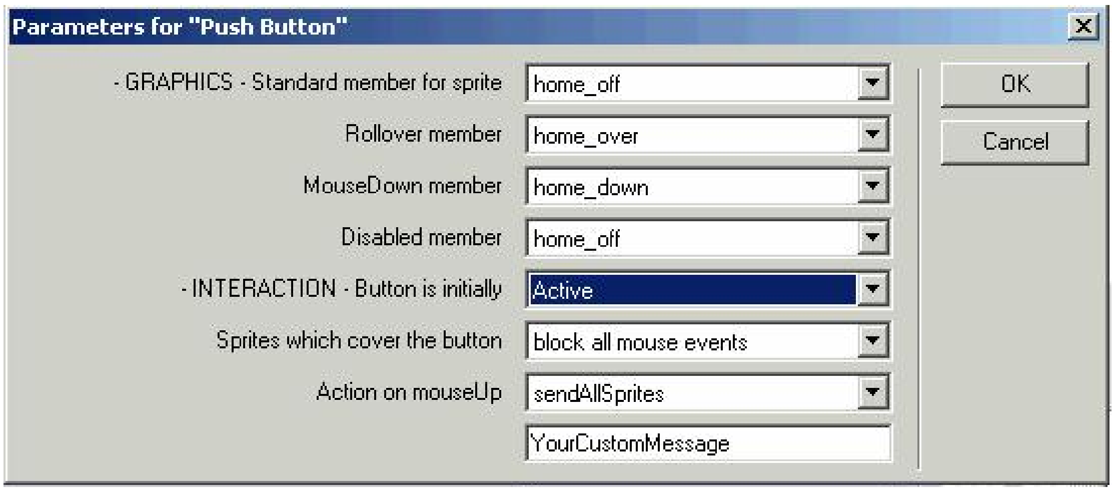


Diagram 17: Dragging the Push button

The **Parameters for "Push Button"** dialogue box is displayed. Observe this dilaogue box. Under graphics, there is a standard Rollover member, Mousedown member and Disabled member reference. The Lingo code in Director has populated these references based on the order of Cast members in the Cast.

- Click on the **Disabled Member** drop-down arrow
- Select **home_off**



Parameters for "Push Button"

- GRAPHICS - Standard member for sprite	home_off
Rollover member	home_over
MouseDown member	home_down
Disabled member	home_off
- INTERACTION - Button is initially	Active
Sprites which cover the button	block all mouse events
Action on mouseUp	sendAllSprites
	YourCustomMessage

Diagram 18: Disabled member

- Click on **OK**
- Close the Library Palette and the Score window
- Click on **Play**

We will move off the Home Page.

- Click on the **6** button

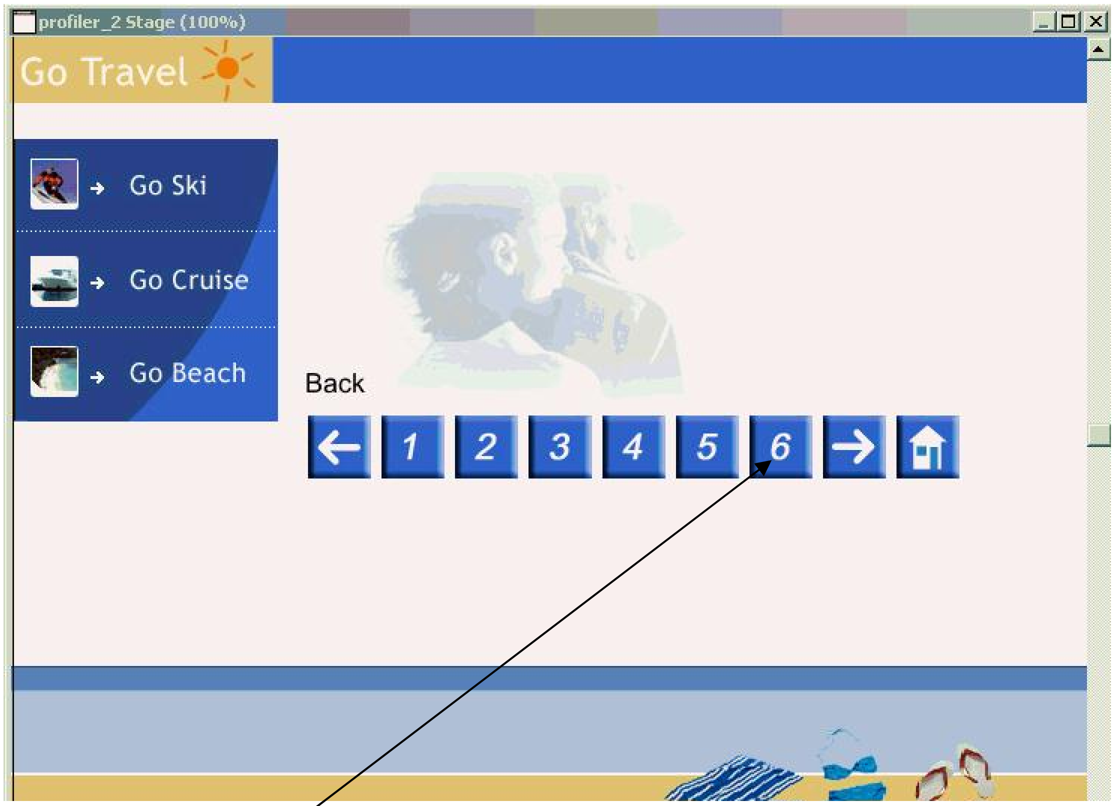


Diagram 19: button 6

Notice that button 6 and all the other buttons except for the Home button, have no mouse event images.

The Home button has a rollover image.

- Click on the **Home** button

The Home button has a Down State image too. All this image swapping is being managed by the Push Button behaviour.

- Stop the movie
- Open the Score and move to Marker **Q1**

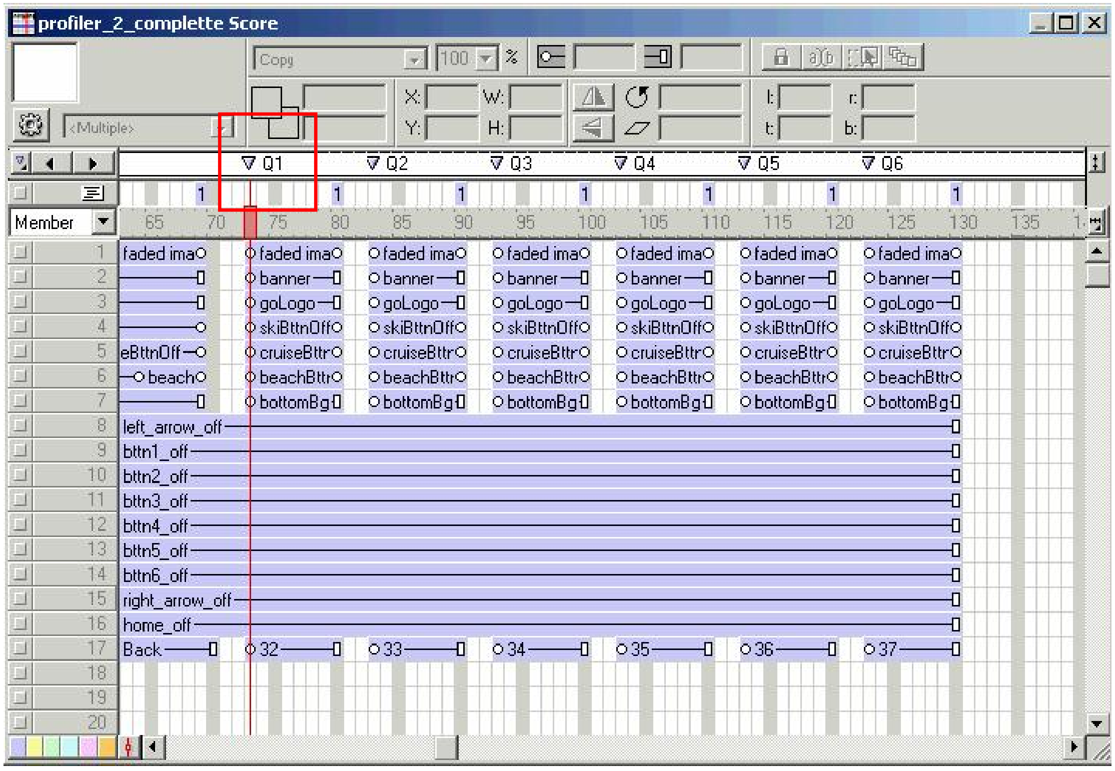
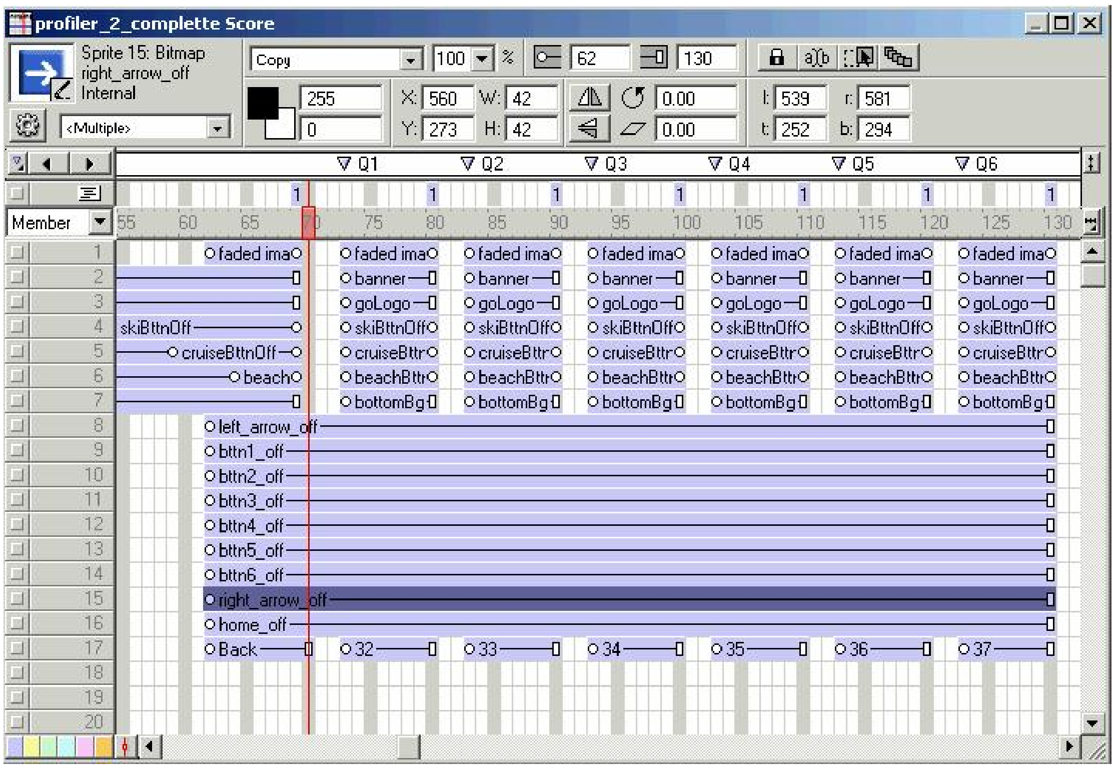


Diagram 20: Marker Q1

- Select the **right_arrow Sprite in channel 15 at frame 70**


profiler_2_complette Score

Sprite 15: Bitmap
right_arrow_off
Internal

Copy 100 % 62 130
255 X: 560 W: 42 0.00 l: 539 r: 581
0 Y: 273 H: 42 0.00 t: 252 b: 294

Diagram 21: Selected Sprite

We will assign the Push Button behaviour to this Sprite.

- Open the Library Palette and drag the **Push Button** behaviour onto Sprite **right_arrow_off** in **channel 15**

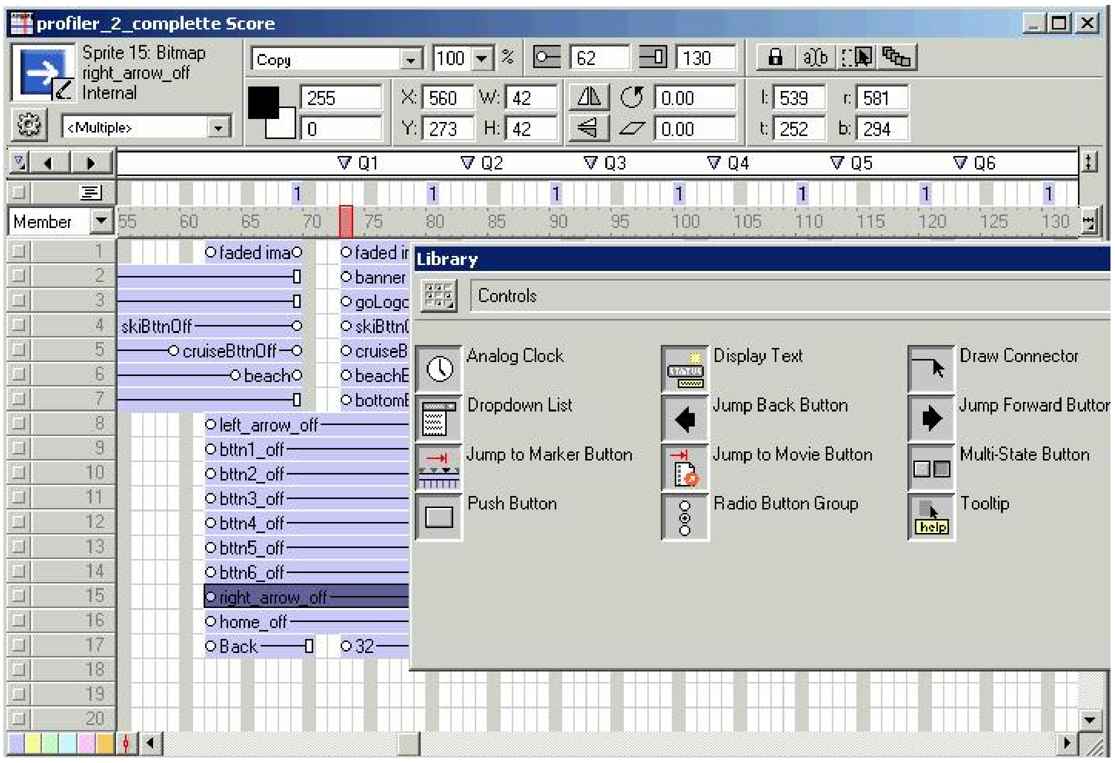


Diagram 22: Dragging the Push button

The parameters for the behaviour have been set for you.

We will look at Sound Behaviours. To explore the Sound Behaviour we will need to import another cast member and place it in the sound channel.

- Import the resource file **funky.wav**
- Place the cast member in **sound channel 1 frame 62**

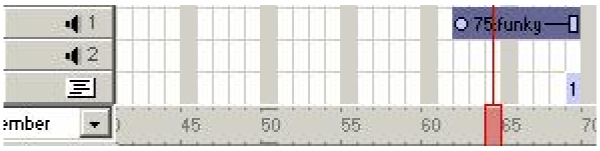


Diagram 23: sound cast member in the sound channle

- Click on the **Library List icon**
- Select **Media>Sound**

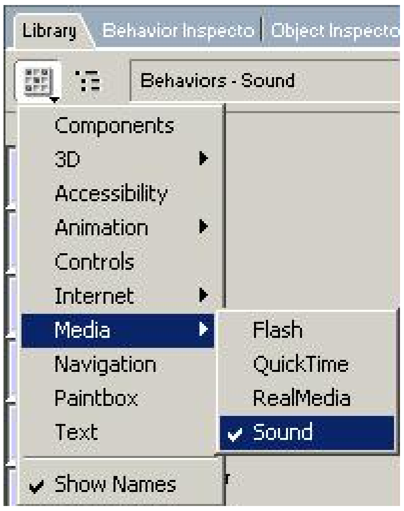


Diagram 24: Media>Sound

The Sound Behaviour options become available on screen.

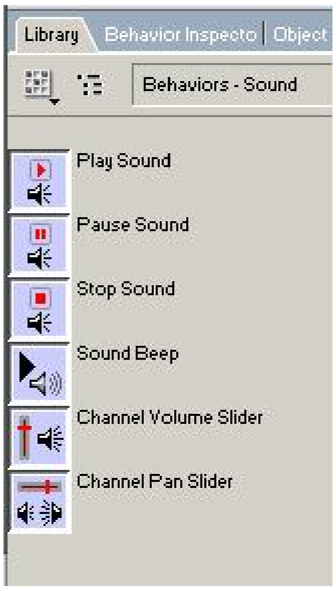
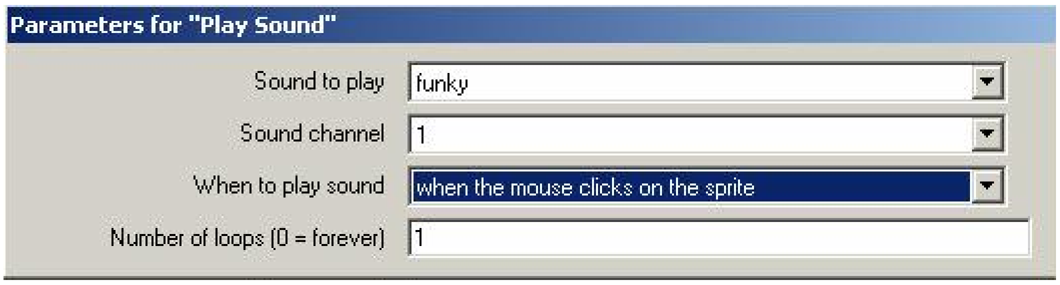


Diagram 25: Sound Behaviour options

- Select **Play Sound**
- Drag this onto the Score and release it on the cast member **Back**

The Parameters for Sound appear on screen.



Parameters for "Play Sound"

Sound to play	funky
Sound channel	1
When to play sound	when the mouse clicks on the sprite
Number of loops (0 = forever)	1

Diagram 26: Parameters for Play Sound dialogue window

- Select **funky** from the sound to play drop down arrow if not already selected
- Set the Sound Channel to 1
- Click on the drop down arrow for the option **"When to play sound"** and select **"when the mouse clicks on the sprite"**
- Set the Number of loops to 1

Take the time to experiment with the sound behaviours as you can apply other settings such as when the sprite first appears or leaves the stage. If you have sound files of your own you may use these to further explore this feature.

The specific properties for each application of the behaviour have been assigned. We will take a closer look. Close down any open windows such as the library to get a better view of your work.

- Click on Rewind
- Click on the **Play** button

The combination of the three behaviours provides your navigation with all the function you identified when you were planning your Lingo scripting.

- Click on the **Stop** button

In this lesson you have learned how:

- To embed Lingo code in a Cast member
- To use the Lingo code in the behaviours that are provided in the Library Palette
- To realise that these behaviours use properties that make it easy to reuse Lingo code